Sprawozdanie

**Analiza**

**Diagram przypadków użycia** (ang. use case diagram) - jest diagramem, który przedstawia funkcjonalność systemu wraz z jego otoczeniem oraz pozwala na graficzne zaprezentowanie własności systemu tak, jak są one widziane po stronie użytkownika.

Pola użyte w diagramie:

* Przypadek użycia – specyfikacja ciągu akcji i ich wariantów, które system (lub inna jednostka) mogą wykonywać poprzez interakcję z aktorami tego systemu.
* Aktor – spójny zbiór ról odgrywanych przez użytkowników przypadku użycia w czasie interakcji z tym przypadkiem użycia.
* Asocjacja – dwukierunkowa komunikacja pomiędzy aktorem a przypadkiem użycia. Może ona występować w formie ze wskazaniem kierunku nawigacji.
* Zależność zawierania „*<<include>>*” – przedstawia powiązanie pomiędzy przypadkiem zawierającym (bazowym przypadkiem użycia), a przypadkiem zawieranym. Zawierany przypadek użycia nie jest wykonywany samodzielnie, lecz wyłącznie przy odwołaniu się do większego, zawierającego przypadku użycia.
* Zależność rozszerzenia „*<<extend>*" – powiazanie pomiędzy rozszerzanym przypadkiem użycia (bazowym przypadkiem użycia), a przypadkiem rozszerzanym.

**Diagram stanu -** reprezentuje zachowanie obiektu o skończonej liczbie stanów i zdefiniowanych przejściach między nimi, jest to graf, który reprezentuje maszynę stanów. Przejście między stanami jest wyzwolone asynchronicznym zdarzeniem zewnętrznym.

Pola użyte w diagramie:

* Stan początkowy – zainicjowanie maszyny stanowej.
* Przejście – relacja między dwoma stanami, wskazująca, że obiekt znajdujący się w pierwszym stanie wykona pewne akcje i przejdzie do drugiego stanu.
* Stan – okoliczność lub sytuacja, w jakiej znajduje się obiekt w cyklu swojego życia, kiedy spełnia warunek, wykonuje czynność lub czeka na zadanie.
* Decyzja – podział przejścia wejściowego na kilka przejść wyjściowych określonych warunkami.
* Punkt zniszczenia – zakończenie maszyny stanowej wskutek zniszczenia adekwatnego obiektu.

**Diagram aktywności** - służy do modelowania czynności i zakresu odpowiedzialności elementów bądź użytkowników systemu. Jest podobny do diagramu stanu, jednak w odróżnieniu od niego nie opisuje działań związanych z jednym obiektem a wieloma, pomiędzy którymi może występować komunikacja przy wykonywaniu czynności.

Pola użyte w diagramie:

* Akcja – elementarna jednostka specyfikacji zachowania, która reprezentuje transformację lub przetwarzanie w modelowaniu systemu.
* Przepływ sterowania – relacja między dwoma akcjami, wskazująca, że po wykonaniu źródłowej akcji sterowanie zostanie przekazanie do docelowej akcji.
* Początek – punkt rozpoczęcie przepływu sterowania.
* Koniec – punkt zatrzymania wszystkich przepływów sterowania.
* Decyzja – podział przejścia wejściowego na kilka wyjściowych określonych warunkami, z których tylko jeden może zostać zrealizowany.
* Łącznik – przerwanie przepływu sterowania i wznowienie go w innym miejscu.
* Rozwidlenie – rozdzielenie jednego wejściowego przepływu na co najmniej dwa wynikowe.
* Scalanie – przekazanie sterowania z wielu współbieżnych, wejściowych przepływów sterowania do jednego wynikowego.

**Diagram wymagań** (ang. requirment diagram) – hierarchiczna lista wymagań. Pozwala na zobrazowanie związków pomiędzy wymaganiami oraz związki wymagań z następnymi elementami projektu.

Pola użyte w diagramie:

* Wymaganie – pojedyncze, proste wymaganie systemowe.
* Zagnieżdżenie – tworzenie wielopoziomowych hierarchii.
* Zależności – wskazują one na charakter logicznej zależności między poszczególnymi wymaganiami. Miedzy innymi zależności realizacji i weryfikowania.
* Testowy przypadek użycia – zawiera algorytm testu.

**Diagram przepływu danych** (ang. Data flow diagram) – przedstawienie w jaki sposób dane przepływają w systemie oraz opis procesów przetwarzających dane.

Pola użyte w diagramie:

* Terminator – obiekt zewnętrzny w stosunku do systemu reprezentujący źródła lub miejsca przeznaczenia informacji.
* Proces – transformacja danych wejściowych w wynikowe.
* Przepływ danych – opisuje zbiór danych przepływających miedzy dwoma obiektami w systemie.
* Magazyn danych – służy do przechowywania danych

Literatura:

https://brasil.cel.agh.edu.pl/~09sbfraczek/

*Język inżynierii systemów SysML. Architektura i zastosowania.* – Stanisław Wrycza, Bartosz Marcinkowski, wydawnictwo Helion, ISBN: 978-83-246-2541-3.

*Język UML 2.0 w programowaniu systemów informatycznych.* - [Stanisław Wrycza](http://helion.pl/autorzy/stanislaw-wrycza,stawry.htm), [Bartosz Marcinkowski](http://helion.pl/autorzy/bartosz-marcinkowski,barmar.htm), [Krzysztof Wyrzykowski](http://helion.pl/autorzy/krzysztof-wyrzykowski,wyrkry.htm), wydawnictwo Helion, ISBN: 83-7361-892-9